เรื่องน่าอ่าน
ลิขสิทธิ์ นำชัย

ระบบจัดการไฟล์ที่ดีกว่า (ดอส)

เปรียบเทียบระบบจัดการไฟล์ระหว่างดอสกับยูนิกซ์เพื่อการเลือกใช้งานอย่างเหมาะสม...

เมื่อบริษัท ไมโครซอฟต์ออกเอ็มเอสดอส 6.0 ออกมานั้น ทำให้หลายคนรู้สึกตื่นเต้น โดยเฉพาะอย่างยิ่งเมื่อดอสเวอร์ชันใหม่นี้มีทั้งคำสั่ง move และ MKS Toolkit ที่ยอมให้ผู้ใช้ที่คุ้นเคยกับระบบยูนิกซ์สามารถใช้คำสั่งของยูนิกซ์บนดอสเวอร์ชันนี้ได้ บางทีโปรแกรมเมอร์ที่ตื่นเต้นกับคำสั่ง move นั้นอาจจะไม่รู้ว่าคำสั่ง move นั้นมีการใช้งานมานานกว่า 20 ปีแล้วในระบบยูนิกซ์ เพียงแต่อยู่ในรูปย่อ mv เพื่อที่ผู้ใช้สามารถเรียกใช้ได้สะดวกขึ้น หลายคนที่เกิดความรู้สึกจำเจในการใช้งานดอสได้หันมาใช้ระบบปฏิบัติการแบบมัลติยูสเซอร์ มัลติทาสกิง มัลติโปรเซสซิง และมัลติเทรดดิ้งแทน ซึ่งช่วยลดปัญหาหลายอย่าง เช่น ข้อจำกัดในการตั้งชื่อไฟล์ หรือการจัดการขนาดของหน่วยความจำสำหรับใช้งาน

ผู้ใช้ทั่วไปนั้นไม่ใช่เป็นเรื่องที่แปลกเลยที่ยึดติดกับดอส เนื่องจากดอสเป็นระบบปฏิบัติการที่ผู้ใช้มีความคุ้นเคย ราคาถูก และมีจำนวนซอฟต์แวร์ที่สนับสนุนการทำงานเป็นจำนวนมาก

แต่สำหรับโปรแกรมเมอร์หรือผู้พัฒนาระบบแล้วทำไมต้องยึดติดกับดอสด้วย แทนที่จะหาระบบปฏิบัติการที่อำนวยความสะดวกในเรื่องต่าง ๆ เช่น การจัดการไฟล์ หรือมีเครื่องมือช่วยในการพัฒนาอย่างระบบยูนิกซ์ซึ่งมีประสิทธิภาพเป็นที่ยอมรับและสามารถทำงานบนเครื่องระดับพีซีได้ อย่างเช่น Solaris 2 ของบริษัท ซัน ไมโครซิสเต็ม

หน่วยความจำเทียม (Vertual Memory)

เมื่อประมาณ 10 ปีที่แล้วโปรแกรมเวิร์ดโปรเซสเซอร์อย่าง Perfect Writer ซึ่งเป็นโปรแกรมที่พัฒนามาจากที่ emac เป็นเอดิเตอร์ของยูนิกซ์ มีการใช้งานกันอย่างแพร่หลายบนระบบปฏิบัติการ CP/M โดยเฉพาะอย่างยิ่งไม่เคยประสบปัญหาเกี่ยวกับขนาดของหน่วยความจำไม่เพียงพอเลย นับว่าเป็นเรื่องที่ยอดเยี่ยมมาก เอ็มเอสดอสเป็นระบบปฏิบัติการที่มีพื้นฐานมาจาก CP/M แต่เรื่องของการจัดการหน่วยความจำยังเป็นรูปแบบที่ค่อนข้างเก่าและไม่สมบูรณ์ ในส่วนของระบบยูนิกซ์นั้นมีการจัดการหน่วย-

ความจำที่ดีกว่า ประกอบกับมียูทิลิตี้เกี่ยวกับการจัดการข้อมูลในดิสค์ อย่างเช่น การบีบอัดข้อมูลให้มีขนาดเล็กลง และอื่น ๆ มีการรวมระบบไฟล์ให้มองเห็นเป็นหนึ่งเดียว และการอ้างถึงนั้นไม่ต้องบอกว่าเป็นไดรฟ์ไหน c: \ แม้ว่าจะเป็นการเก็บบนดิสค์หรือซีดีรอมก็ตาม แม้กระทั่งข้อมูลที่อยู่ในเครื่องอื่นที่อาศัยการเรียกผ่านระบบเครือข่าย ภายใต้ระบบยูนิกซ์มีการอ้างถึงหลายรูปแบบ โดยเฉพาะการอ้างไฟล์แบบหลังสุดนั้นเป็นแบบระบบไฟล์กระจาย (Distributed File System) ที่มีโปรแกรมยูทิลิตี้ที่ชื่อ NFS (Network File System) ที่ทำงานในระดับชั้นเหนือโปรโตคอล TCP/IP และลักษณะของโปรโตคอลนี้เป็นเสมือนมาตรฐานที่ต้องมีในยูนิกซ์ทุกระบบแล้ว

การจัดการไฟล์

คำสั่ง mv ไม่เพียงแต่เข้าถึงไฟล์โดยการเปลี่ยนชื่อเท่านั้น ไม่มีการย้ายหรือสำเนาข้อมูล การเชื่อมข้อมูลทำได้ด้วยคำสั่ง ln (link) เช่น ln original__name new__name เป็นการเชื่อมไฟล์จริงกับชื่อไฟล์ใหม่ในไดเรคทอรีใหม่ที่ต้องการ อาจเป็นที่สงสัยว่าการเชื่อมเช่นนี้มีประโยชน์อย่างไร ตัวอย่างเช่น ระบบที่มีไฟล์เป็นจำนวนมากจำเป็นต้องมีการจัดไฟล์ออกเป็นหมวดหมู่อย่างไฟล์ หรือโปรแกรมที่รันได้จะอยู่ในไดเรคทอรีหนึ่ง เช่น /usr/bin เนื้อที่ว่างของไดเรคทอรีนี้อาจมีเหลืออยู่น้อยมากจนไม่สามารถเพิ่มเติมไฟล์ลงไปได้อีก หากมีโปรแกรมประยุกต์บางโปรแกรมที่มีลักษณะการติดตั้งบังคับว่าโปรแกรมประยุกต์นั้นจะสามารถทำงานได้ก็ต่อเมื่อติดตั้งในไดเรคทอรี /usr/bin เท่านั้น คำสั่ง ln จะช่วยให้สามารถติดตั้งโปรแกรมประยุกต์นั้นไว้ในเนื้อที่ว่างส่วนไหนก็ได้ แล้วทำการเชื่อมโปรแกรมนั้นมาที่ยังไดเรคทอรีที่ต้องการ โดยที่โปรแกรมประยุกต์นั้นสามารถทำงานได้ตามปกติและมองไม่เห็นถึงความแตกต่าง แต่การเชื่อมที่เป็นการเชื่อมข้ามอุปกรณ์นั้น คำสั่ง ln ไม่สามารถทำได้ สำหรับในกรณีนี้ต้องใช้การเชื่อมต่อแบบพิเศษที่เป็น Symbolic Link โดยการเพิ่มออปชัน -s หลังคำสั่ง ln การเข้าถึงไฟล์นั้นจะเป็นลักษณะพิเศษ และสิ่งที่น่าสนใจของ Symbolic Link คือ การที่สามารถเชื่อมเข้าไปยังไฟล์ที่ยังไม่มีจริงได้ เช่น ไฟล์ที่อยู่บนดิสค์ในขณะที่ยังไม่มีการเม้าต์ (mount) ไฟล์ซิสเต็มส์นั้นเข้ามา

การแสดงรายชื่อไฟล์นั้นสามารถกระทำได้โดยใช้คำสั่ง ls (เทียบเท่ากับคำสั่ง DIR) รูปแบบการแสดงผลจากคำสั่งของยูนิกซ์นั้นถูกออกแบบให้สามารถใช้งานร่วมกับคำสั่งอื่นได้ อย่างเช่น ผู้ใช้สามารถเรียกแสดงรายชื่อไฟล์พร้อมกับส่งผลลัพธ์ที่ได้ให้กับโปรแกรม makefile ในการรวมไฟล์ต่าง ๆ เข้าด้วยกันและทำการคอมพายล์ต่อได้ทันที ที่เป็นเช่นนี้เพราะผลลัพธ์ในยูนิกซ์เป็นการแสดงเฉพาะส่วนที่ต้องการ (ดังรูปที่ 1) หากต้องการส่วนอื่นเพิ่มก็ต้องเติมออปชันเข้าไป ส่วนการแสดงรายชื่อไฟล์ของดอสนั้นมีส่วนเกินความต้องการออกมามาก ไม่ว่าจะเป็น ลาเบลของดิสค์ ขนาดของไฟล์ วันเวลาที่สร้างหรือแก้ไข จำนวนไฟล์ ขนาดของไฟล์รวม และเนื้อที่ว่างที่เหลืออยู่ในอุปกรณ์ที่เก็บไฟล์เหล่านั้น ดังรูปที่ 2

รูปที่ 1 คำสั่ง ls แสดงรายชื่อไฟล์ในระบบยูนิกซ์

```
$ ls
lession.c   lession2.c   lession3.c   lession4.c
lession7.c  mbox         a.out
lession5.c  lession6.c
```

```
Volume in drive C is Likhasith Serial number is DD9:2965
Directory of c: \ rick \ *.*
.          <DIR>      6-17-93 5:22a
..         <DIR>      6-17-93 5:22a
lesson2.c          2406   2-01-93-12:00p
normal.c             40   9-19-93 5.07p
test               2446   2-01-93 12:00p
users.c           18083   5-27-93 3:21p
utitle.c           2445   5-27-93 6:09p
        25,420 bytes in 7 file(s)        49,152 bytes allocated
15,507,456 bytes free
```

รูปที่ 2 คำสั่ง DIR ในการแสดงรายชื่อไฟล์ของดอส

การค้นหาไฟล์ในดอสนั้นนับได้ว่าเป็นอีกเรื่องหนึ่งที่มีปัญหา สำหรับในระบบยูนิกซ์นั้นได้เตรียมคำสั่ง find ในการค้นหาไฟล์ไว้ และมีเงื่อนไขช่วยในการค้นหาหลายรูปแบบ โดยอาจเป็นชื่อไฟล์ ชนิดของไฟล์ ช่วงเวลาก่อนหรือหลังที่มีการสร้างหรือเปลี่ยนแปลงแก้ไขไฟล์ การเป็นเจ้าของหรือกลุ่มของเจ้าของไฟล์ สิทธิในการจัดการไฟล์ ลำดับการค้นจะค้นจากไดเรกทอรีตั้งต้นที่กำหนดลงมาตามลำดับชั้น หากพบรูปแบบที่ตรงกับเงื่อนไขก็แสดงตำแหน่งที่อยู่ของไฟล์นั้นเป็นพาธ (Path) ออกมา และในกรณีที่ผู้ใช้รู้เพียงอักษรบางตัวของชื่อไฟล์ก็สามารถใช้สัญลักษณ์พิเศษช่วยได้ พร้อมทั้งยังสามารถผ่านค่าให้กับคำสั่งอื่น อย่างเช่น cpio (Copy In Copy Out) เพื่อทำการสำรองข้อมูลไปยังอุปกรณ์ที่ต้องการได้ นับว่าเป็นประโยชน์กับผู้ดูแลระบบที่สามารถกำหนดเงื่อนไขในการเลือกสำรองข้อมูลได้ สำหรับการค้นหาแบบหยาบ ๆ สามารถใช้คำสั่ง ls -R ที่แสดงรายชื่อไฟล์ในไดเรกทอรีย่อย อย่างเช่น ls -R / เป็นการแสดงรายชื่อไฟล์ทั้งหมด

ในกรณีที่ผู้ใช้นั้นไม่ทราบข้อมูลอะไรเกี่ยวกับไฟล์นั้น รู้แต่เพียงว่าข้อมูลภายในมีตัวแปรว่าอย่างไรบ้าง ก็สามารถทำการค้นหาได้โดยใช้คำสั่ง grep เช่น

```
grep variable -name *.c
```

จะแสดงข้อมูลทั้งหมด และถ้าผู้ใช้เพิ่มออปชัน –n ก็จะแสดงหมายเลขที่บรรทัดให้ด้วย และในกรณีที่มีการสับสนในเรื่องของชนิดตัวอักษรว่าเป็นตัวเล็กหรือตัวใหญ่นั้นสามารถไม่สนใจชนิดของตัวอักษรได้ โดยใช้ออปชัน -i หรือถ้าหากต้องการรู้ถึงจำนวนบรรทัดและตัวอักษรนั้นก็ทำได้ง่าย โดยผ่านค่าให้กับคำสั่ง wc (-l แสดงจำนวนบรรทัด -c แสดงจำนวนตัวอักษร -w แสดงจำนวนคำ)

แต่ระบบยูนิกซ์ก็มีปัญหาอยู่เหมือนกันในเรื่องของออปชันที่มีความหมายไม่สอดคล้องกันในทุกคำสั่ง อย่างเช่น ออปชัน -r ของคำสั่ง rm (remove) เป็นลักษณะการทำซ้ำ (recursive) แต่ -r ในคำสั่ง cp (Copy) เป็นการทำย้อนกลับ (Reverse) ส่วนการทำซ้ำนั้นเป็นออปชัน -R แทน เหตุที่ยูนิกซ์เป็นเช่นนี้เนื่องจากระบบยูนิกซ์เป็นผลิตผลมาจากหลาย ๆ บริษัท แตกต่างกับดอสที่มีรูปแบบคำสั่งเฉพาะแน่นอน

ยูนิกซ์บนเครื่องระดับพีซี

ก่อนหน้านี้เมื่อเอ่ยถึงระบบยูนิกซ์บนพีซีใคร ๆ ก็ต้องคิดว่าเป็น XENIX เพราะเป็นผลิตภัณฑ์ที่ประสบความสำเร็จทางการค้า ส่วนระบบยูนิกซ์บนพีซีตัวอื่น ๆ อย่าง VENIX, PC/IX และ Microport มักถูกมองข้ามไป ซึ่งเหตุที่เป็นเช่นนี้เกิดจากข้อจำกัดของพีซีในเรื่องของทรัพยากรต่าง ๆ ที่มีความสามารถต่ำเมื่อเทียบกับมินิคอมพิวเตอร์ ดังนั้นผู้ค้าระบบยูนิกซ์จึงมักทำการแยกจำหน่ายยูนิกซ์ออกเป็น 3 ส่วนคือ ระบบพื้นฐาน (Basic) ส่วนประมวลผลตัวอักษร (Text Processing) และส่วนของการโปรแกรม (Programming) และการจัดการของระบบยูนิกซ์มีการทำงานที่ซับซ้อน ขณะที่พีซีที่รองรับนั้นการติดต่อกับดิสก์ทำงานช้าและมีขนาดของหน่วยความจำจำกัด จึงทำให้โปรแกรมที่ทำงานบนยูนิกซ์ทำงานได้ช้าเมื่อเทียบกับการทำงานบนดอส

บริษัทที่ค้ายูนิกซ์บนเครื่องระดับพีซีได้มีการปรับปรุงพัฒนาให้ดีขึ้นเป็นลำดับ และส่วนของผู้ผลิตที่เป็นที่รู้จักมีอยู่ 3 บริษัทคือ AT&T ที่ผลิต System/386 บริษัท Santa Cruz Operation ผลิต SCO โดยมีการเพิ่มรูปแบบการติดต่อ ระบบเมนูในการดูแลระบบ (System administration Shell) และระบบ X Windows รวมเรียกว่า Open Desktop ส่วนอีกบริษัทคือ บริษัท ซัน ไมโครซิสเต็ม ที่ได้นำข้อดีของยูนิกซ์ที่ชื่อว่า Interactive มาเพิ่มประสิทธิภาพแก่ Solaris 2 ทางผู้ผลิตหวังว่าจะเป็นระบบปฏิบัติการที่มีความสามารถสูงสุดและข้อจำกัดน้อยสุด โดยมีรูปแบบใน SVR4 ตามมาตรฐานของบริษัท ซัน ไมโครซิสเต็ม นับว่าเป็นการทุ่มเทและลงทุนครั้งใหญ่ของบริษัท ซัน ไมโครซิสเต็ม ในการพัฒนาระบบยูนิกซ์ในระดับพีซี แต่ก็ยังประสบปัญหาอยู่ 2 ประการคือ จำนวนซอฟต์แวร์ของผู้ค้าอื่น (Third party) ยังมีน้อย แต่คาดว่าคงจะเพิ่มได้ตามความต้องการ และอีกปัญหาคือ แม้จะสอดคล้องกับ SVR4 แต่ก็มีรูปแบบที่แตกต่างไปจาก SVR4 อื่น แต่ปัญหานี้จะหมดสิ้นเมื่อมีการใช้มาตรฐาน COSE (Common Open Software Environment)

ส่งท้าย

การแข่งขันของระบบปฏิบัติการในเครื่องหลายระดับเต็มไปด้วยความเข้มข้น ในเครื่องระดับพีซีก็เช่นเดียวกัน และเมื่อบริษัท ซัน ไมโครซิสเต็มที่เป็นส่วนแบ่งตลาดรายใหญ่ในเครื่องระดับบนได้หันมาสนใจและพัฒนายูนิกซ์บนเครื่องในระดับพีซี จึงเกิดคำถามที่ว่า

ใกล้ถึงเวลาที่จะเลิกใช้งานดอสแล้วหรือ?